UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/832,373	04/11/2001	Tonis Kasvand	8673-117 (8061-598 SJP:kl	4525

22150 7590 11/27/2006

F. CHAU & ASSOCIATES, LLC
130 WOODBURY ROAD
WOODBURY, NY 11797

EXAMINER	
PATEL, ASHOKKUMAR B	

ART UNIT	PAPER NUMBER
2154	

DATE MAILED: 11/27/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/832,373	**Applicant(s)** KASVAND ET AL.
	Examiner Ashok B. Patel	**Art Unit** 2154

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _07 September 2006_.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-7_ is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-7_ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _6/28/2004_.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-7 are presented for examination.

2. This action is responsive to appeal brief filed on 09/07/2006. In view of the appeal brief filed on 09/07/2006, PROSECUTION IS HEREBY REOPENED. New grounds of rejections are set forth below.

To avoid abandonment of the application, appellant must exercise one of the following two options:

(1) file a reply under 37 CFR 1.111 (if this Office action is non-final) or a reply under 37 CFR 1.113 (if this Office action is final); or,

(2) request reinstatement of the appeal.

If reinstatement of the appeal is requested, such request must be accompanied by a supplemental appeal brief, but no new amendments, affidavits (37 CFR 1.130, 1.131 or 1.132) or other evidence are permitted. See 37 CFR 1.193(b)(2).

3. Examiner would like to thank the Applicant for providing the summary on the claimed subject matter, and explanation of the claimed limitations in relation to the specification by reference to Figures, pages and lines. Upon further consideration given to the understanding of the claimed limitations, new grounds of rejection is made as below.

Claim Rejections - 35 USC § 112

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. Claims 1-7 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Referring to claim 1,

Claim 1 recites in preamble "A network administration system for triggering commands in response to receipt of status logs", however, the claim further recites the limitation followed by the preamble "receiving said higher level logs, parsing each of said predetermined ones of said higher level logs to determine their respective sources, and trigging execution of said commands in said execution sets in respect of each of said respective sources." The preamble and the later claim limitation are unclear and unable to establish the relation to "triggering commands in response to receipt of status logs".

For the purpose of this office action, although the following rejection shows the claimed limitations taught by the appropriate prior art, preamble is not given the consideration as to resulting into any relation to the later claim limitations for the reason stated above. And as preamble is not given the consideration, the "receiving said status logs" is not considered to have any relation to "trigging execution of said commands".

Referring to claims 2, 3 and 6,

Claims 2, 3 and 6 are also rejected for the reasons set forth for claim 1 because of their dependency on claim 1.

Referring to claim 4,

Claim 4 recites in preamble "A method of triggering commands in response to receipt of status logs", however, the claim further recites the claim limitation followed by the preamble "receiving said status logs and higher level logs, parsing each of said predetermined ones of said higher level logs to determine their respective sources, and trigging execution of said commands in said execution sets in respect of each of said respective sources." The preamble and the later claim limitation are unclear and unable to establish the relation to "triggering commands in response to receipt of status logs"

For the purpose of this office action, although the following rejection shows the claimed limitations taught by the appropriate prior arts, preamble is not given the consideration as to resulting into any relation to the later claim limitations for the reason stated above. And as preamble is not given the consideration, the "receiving said status logs" is not considered to have any relation to "trigging execution of said commands".

Referring to claims 5 and 7,

Claims 5 and 7 are also rejected for the reasons set forth for claim 4 because of their dependency on claim 4.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 1-7 are rejected under 35 U.S.C. 103(a) as being Unpatentable over Cuddihy et al. (hereinafter Cuddihy) (US 5, 463, 768) in view of Noble et al. (hereinafter Noble) (US 5, 944, 782)

Referring to claim 1,

Cuddihy teaches a network administration system for receiving status logs generated by network devices and applications (col. 3, line 53-59, "The sources include: field repair sites, central repair facility, quality control testing laboratories, etc. The plurality of error logs are then used as historical cases documenting software and hardware errors occurring at the different machines. Each of the error logs has a corresponding malfunction description (i.e. fault nz, yw, bd, etc.) associated with it."), comprising:

means for receiving said status logs (col. 3, line 49-50, "The training unit 14 receives the plurality of error logs from various imaging devices located at different locations.") and generating higher level logs in response to receipt of at least two different status logs (col. 4, line 33-37, "After formatting, each of the plurality of error logs 30 are grouped in the block finding and matching unit 32 into case sets of common symptoms or common corrective actions (i.e. faulty parts, boards, caches, etc.).") which satisfy predetermined rule sets (col. 4, line 33-50, "After formatting, each of the plurality of error logs 30 are grouped in the block finding and matching unit 32 into case sets of common symptoms or common corrective actions (i.e. faulty parts, boards, caches, etc.). FIG. 5A shows error logs 1-11 grouped into case sets, wherein error log cases 1-3 are grouped into case set I; error log cases 4-5 into case set II; error log cases 6-9 into case set III; and error log cases 10-11 into case set IV. A case is represented by one or more error logs and a fix associated with a single malfunction of a device. A case set consists of all historical cases having the same fix. After the error logs have been grouped into cases and case sets, the error logs in each case set are examined to identify common patterns of data. The common patterns are then labelled as blocks. A block consists of consecutive row(s) of data in a data file such as represented by FIGS. 4A-4B derived from a historical error log file that exists in at least one or more cases." Please note that error logs are status logs and "a case" is a "higher level log");

Cuddihy also teaches program means for receiving said higher level logs, parsing each of said predetermined ones of said higher level logs to determine their respective sources (col. 4, line 44-59, "After the error logs have been grouped into cases and case sets, the error logs in each case set are examined to identify common patterns of data (program means for receiving said higher logs and parsing and each case is predetermined higher level log). The common patterns are then labelled as blocks. A block consists of consecutive row(s) of data in a data file such as represented by FIGS. 4A-4B derived from a historical error log file that exists in at least one or more cases. FIG. 5B shows the error logs in case set I having blocks A, B, G and H; the error logs in case set II having blocks A, B, C and E; the error logs in case set III having blocks A, B, C, D, and F; and the error logs in case set IV having blocks A, B, C, D, E, and F. After blocks in each case set have been identified, the blocks of each case set are compared to identify common blocks." Please also note as stated above that "A case is represented by one or more error logs and a fix associated with a single malfunction of a device. A case set consists of all historical cases having the same fix.", as such the sources of error is also known);

Although Cuddihy gives silent indication of interface by stating "Thus, whether to use or not use parsing or columnizing depends on the user." (col. 4, line 31-32), Cuddihy explicitly fails to teach triggering commands in response to receipt of logs, a user interface for programming execution sets of commands in association with predetermined ones of said higher level logs; and trigging execution of said commands in said execution sets in respect of each of said respective sources.

Noble teaches triggering commands in response to receipt of logs (col.2, line 22-25), a user interface (col. 1, line 64, col. 2, line 1, "A management interface is a program executing on a computer (referred to as an "administrative" computer) with which a system administrator interacts to monitor and direct management operations across managed computers.") for programming execution sets of commands (col. 2, line 32-33, "These routines can be scripts prescribed by the system administrator.") in association with predetermined ones of said logs (col. 2, line 40-42, "Alternatively, the corrective scripts can be run by management agents as remote shell execution on a management engine or management interface itself. In addition, the management system provides for filtering of alarm messages from various managed computers to avoid redundancy and false alarms."); and trigging execution of said commands in said execution sets in respect of each of said respective sources (col. 5, line 33-39, "Each alarm 590 may also include one or more corresponding scripts 598 which are responsive or corrective processes initiated by the alarm 590 in response to the occurrence of a predefined event. Each event definition may be applied to multiple managed computers 330. Once an alarm 590 is triggered it persists until it is reset. Each set of attributes 596 and each script 598 may be specified by the system administrator.")

Cuddihy teaches it's invention as being "providing an error log analysis system that automatically generates diagnostic knowledge without the dependence of human experts such as field engineers."(col.2, line 25-28), and provides the solution to extract important parameters from case-based diagnostic system and finds applicable solution, however, does not provide any means to automatically generate corrective action execution as taught by Noble as stated above.

Therefore it would have been an obvious to one of an ordinary skill in art, having the teachings of Cuddihy and Noble in front of him at the time of invention was made, to include the teachings of Noble into the Cuddihy's case-based diagnostic system incorporating cases (higher level logs) such that the case-based diagnostic system automatically generates corrective action execution (triggering execution of commands) wherein the corrective routines (scripts- execution sets) are responsive or corrective processes initiated by the logs in response to the occurrence of a predefined event that be applied to multiple managed devices as prescribed by the system administrator (user).

This would have been obvious because Noble enhances the Cuddihy's system as it would not only automatically generate corrective action execution (triggering execution of commands) but also provides flexibility such as the corrective scripts (execution sets) can be run as remote shell execution on a management engine or management interface itself, one or more corresponding scripts which are responsive or corrective processes initiated by the alarm (status log) in response to the occurrence of a predefined event, and each event definition may be applied to multiple managed computers (devices), which boils down to providing a management system that is sufficiently flexible to handle the administration of a wide variety and a large number of managed computers (devices) in an efficient manner, essentially preventing the systems administrator from becoming overwhelmed with the increasing amount of management information and action required, and to automate these operations as much as possible, minimize the number of decisions and actions that are required by the system administrator, and a management system that is optionally self-configuring, as Noble puts it (col. 1, line 38-54).

Referring to claims 2 and 3,

Keeping in mind the teachings of Cuddihy as stated above, including silent indication of interface by stating "Thus, whether to use or not use parsing or columnizing depends on the user." (col. 4, line 31-32), Cuddihy fails to teach network administration system of claim 1, wherein said user interface provides ordered execution of multiple commands associated with said higher level logs in accordance with user preference, and the network administration system of claim 1, wherein said user interface and program means are implemented within one of said network devices.

Noble teaches a user interface (col. 1, line 64, col. 2, line 1, "A management interface is a program executing on a computer (referred to as an "administrative" computer) with which a system administrator interacts to monitor and direct management operations across managed computers.") provides ordered execution of multiple commands associated with logs (col. 5, line 33-39, "Each alarm 590 may also include one or more corresponding scripts 598 which are responsive or corrective processes initiated by the alarm 590 in response to the occurrence of a predefined event. Each event definition may be applied to multiple managed computers 330. Once an alarm 590 is triggered it persists until it is reset. Each set of attributes 596 and each script 598 may be specified by the system administrator.") in accordance with user preference, (col. 2, line 32-33, "These routines can be scripts prescribed by the system administrator.", col. 2, line 40-42, "Alternatively, the corrective scripts can be run by management agents as remote shell execution on a management engine or management interface itself.);

Cuddihy teaches it's invention as being "providing an error log analysis system that automatically generates diagnostic knowledge without the dependence of human experts such as field engineers."(col.2, line 25-28), and provides the solution to extract important parameters from case-based diagnostic system and finds applicable solution, however, does not provide any means to automatically generate corrective action execution as taught by Noble as stated above.

Therefore it would have been an obvious to one of an ordinary skill in art, having the teachings of Cuddihy and Noble in front of him at the time of invention was made, to include the teachings of Noble into the Cuddihy's case-based diagnostic system incorporating cases (higher level logs) such that the case-based diagnostic system automatically generates corrective action execution (triggering execution of commands) wherein the corrective routines (scripts- execution sets) are responsive or corrective processes initiated by the logs in response to the occurrence of a predefined event that be applied to multiple managed devices as prescribed by the system administrator (user).

This would have been obvious because Noble enhances the Cuddihy's system as it would not only automatically generate corrective action execution (triggering execution of commands) but also provides flexibility such as the corrective scripts (execution sets) can be run as remote shell execution on a management engine or management interface itself, one or more corresponding scripts which are responsive or corrective processes initiated by the alarm (status log) in response to the occurrence of a predefined event, and each event definition may be applied to multiple managed computers (devices), which boils down to providing a management system that is sufficiently flexible to handle the administration of a wide variety and a large number of managed computers (devices) in an efficient manner, essentially preventing the systems administrator from becoming overwhelmed with the increasing amount of management information and action required, and to automate these operations as much as possible, minimize the number of decisions and actions that are required by the system administrator, and a management system that is optionally self-configuring, as Noble puts it (col. 1, line 38-54).

Referring to claim 4,

Coddihy teaches a method of receipt of status logs generated by network devices and applications (col. 3, line 53-59, "The sources include: field repair sites, central repair facility, quality control testing laboratories, etc. The plurality of error logs are then used as historical cases documenting software and hardware errors occurring at the different machines. Each of the error logs has a corresponding malfunction description (i.e. fault nz, yw, bd, etc.) associated with it."), comprising the steps of:

providing rule sets that are satisfied (col. 4, line 33-50, "After formatting, each of the plurality of error logs 30 are grouped in the block finding and matching unit 32 into case sets of common symptoms or common corrective actions (i.e. faulty parts, boards, caches, etc.). FIG. 5A shows error logs 1-11 grouped into case sets, wherein error log cases 1-3 are grouped into case set I; error log cases 4-5 into case set II; error log cases 6-9 into case set III; and error log cases 10-11 into case set IV. A case is represented by one or more error logs and a fix associated with a single malfunction of a device. A case set consists of all historical cases having the same fix. After the error logs have been grouped into cases and case sets, the error logs in each case set are examined to identify common patterns of data. The common patterns are then labelled as blocks. A block consists of consecutive row(s) of data in a data file such as represented by FIGS. 4A-4B derived from a historical error log file that exists in at least one or more cases.") by receipt of particular combinations of at least two different status logs (col. 3, line 49-50, "The training unit 14 receives the plurality of error logs from various imaging devices located at different locations.") and when satisfied, result in the generation of higher level logs(col. 4, line 33-37, "After formatting, each of the plurality of error logs 30 are grouped in the block finding and matching unit 32 into case sets of common symptoms or common corrective actions (i.e. faulty parts, boards, caches, etc.)." Please note that error logs are status logs and "a case" is a "higher level log");

Cuddihy also teaches receiving said status logs and said higher level logs, parsing each of said predetermined ones of said higher level logs to determine their respective sources (col. 4, line 44-59, "After the error logs have been grouped into cases and case sets, the error logs in <u>each case set are examined to identify common patterns of data</u>. The common patterns are then labelled as blocks. A block consists of consecutive row(s) of data in a data file such as represented by FIGS. 4A-4B derived from a historical error log file that exists in at least one or more cases. FIG. 5B shows the error logs in case set I having blocks A, B, G and H; the error logs in case set II having blocks A, B, C and E; the error logs in case set III having blocks A, B, C, D, and F; and the error logs in case set IV having blocks A, B, C, D, E, and F. After blocks in each case set have been identified, the blocks of each case set are compared to identify common blocks." Please also note as stated above that "<u>A case is represented by one or more error logs and a fix associated with a single malfunction of a device. A case set consists of all historical cases having the same fix.</u>", as such the sources of error is also known);

Although Cuddihy gives silent indication of interface by stating "Thus, whether to use or not use parsing or columnizing depends on the user." (col. 4, line 31-32), Cuddihy explicitly fails to teach triggering commands in response to receipt of logs , <u>programming execution sets</u> of commands in association with predetermined ones of said higher level logs; and trigging execution of said commands in said execution sets in respect of each of said respective sources.

Noble teaches triggering commands in response to receipt of logs (col.2, line 22-25), a user interface (col. 1, line 64, col. 2, line 1, "A management interface is a program executing on a computer (referred to as an "administrative" computer) with which a system administrator interacts to monitor and direct management operations across managed computers.") for programming execution sets of commands (col. 2, line 32-33, "These routines can be scripts prescribed by the system administrator.") in association with predetermined ones of said logs (col. 2, line 40-42, "Alternatively, the corrective scripts can be run by management agents as remote shell execution on a management engine or management interface itself. In addition, the management system provides for filtering of alarm messages from various managed computers to avoid redundancy and false alarms."); and trigging execution of said commands in said execution sets in respect of each of said respective sources (col. 5, line 33-39, "Each alarm <u>590 may also include one or more corresponding scripts</u> 598 which are responsive or corrective processes initiated by the alarm 590 in response to the occurrence of a predefined event. Each event definition may be <u>applied to multiple managed computers</u> 330. Once an alarm 590 is triggered it persists until it is reset. Each set of attributes 596 and each script 598 may be specified by the system administrator.")

Cuddihy teaches it's invention as being "providing an error log analysis system that automatically generates diagnostic knowledge without the dependence of human experts such as field engineers."(col.2, line 25-28), and provides the solution to extract important parameters from case-based diagnostic system and finds applicable solution, however, does not provide any means to automatically generate corrective action execution as taught by Noble as stated above.

Therefore it would have been an obvious to one of an ordinary skill in art, having the teachings of Cuddihy and Noble in front of him at the time of invention was made, to include the teachings of Noble into the Cuddihy's case-based diagnostic system incorporating cases (higher level logs) such that the case-based diagnostic system automatically generates corrective action execution (triggering execution of commands) wherein the corrective routines (scripts- execution sets) are responsive or corrective processes initiated by the logs in response to the occurrence of a predefined event that be applied to multiple managed devices as prescribed by the system administrator (user).

This would have been obvious because Noble enhances the Cuddihy's system as it would not only automatically generate corrective action execution (triggering execution of commands) but also provides flexibility such as the corrective scripts (execution sets) can be run as remote shell execution on a management engine or management interface itself, one or more corresponding scripts which are responsive or corrective processes initiated by the alarm (status log) in response to the occurrence of a predefined event, and each event definition may be applied to multiple managed computers (devices), which boils down to providing a management system that is sufficiently flexible to handle the administration of a wide variety and a large number of managed computers (devices) in an efficient manner, essentially preventing the systems administrator from becoming overwhelmed with the increasing amount of management information and action required, and to automate these operations as much as possible, minimize the number of decisions and actions that are required by the system administrator, and a management system that is optionally self-configuring, as Noble puts it (col. 1, line 38-54).

Referring to claim 5,

Cuddihy teaches the method of claim 4, wherein said step of receiving said status logs and said higher level logs, parsing each of said predetermined ones of said higher level logs to determine their respective sources(col. 4, line 44-59, "After the error logs have been grouped into cases and case sets, the error logs in each case set are examined to identify common patterns of data. The common patterns are then labelled as blocks. A block consists of consecutive row(s) of data in a data file such as represented by FIGS. 4A-4B derived from a historical error log file that exists in at least one or more cases. FIG. 5B shows the error logs in case set I having blocks A, B, G and H; the error logs in case set II having blocks A, B, C and E; the error logs in case set III having blocks A, B, C, D, and F; and the error logs in case set IV having blocks A, B, C, D, E, and F. After blocks in each case set have been identified, the blocks of each case set are compared to identify common blocks." Please also note as stated above that "A case is represented by one or more error logs and a fix associated with a single malfunction of a device. A case set consists of all historical cases having the same fix.", as such the sources of error is also known).

Although Cuddihy gives silent indication of interface by stating "Thus, whether to use or not use parsing or columnizing depends on the user." (col. 4, line 31-32), Cuddihy explicitly fails to teach triggering execution of said commands in said execution sets further comprise the steps of: a) detecting an execution set associated with a received higher level log; and b) executing each successive commands in said execution set.

Noble teaches triggering execution of said commands in said execution sets (col. 2, line 32-33, "These routines can be scripts prescribed by the system administrator.") (col. 2, line 40-42, "Alternatively, the corrective scripts can be run by management agents as remote shell execution on a management engine or management interface itself. In addition, the management system provides for filtering of alarm messages from various managed computers to avoid redundancy and false alarms."); and detecting an execution set associated with a received higher level log (col. 8, line 41-44, "If no auto-executing script 596 exists".. and b) executing each successive commands in said execution set. (col. 5, line 33-39, "Each alarm 590 may also include one or more corresponding scripts 598 which are responsive or corrective processes initiated by the alarm 590 in response to the occurrence of a predefined event. Each event definition may be applied to multiple managed computers 330. Once an alarm 590 is triggered it persists until it is reset. Each set of attributes 596 and each script 598 may be specified by the system administrator.")

Cuddihy teaches it's invention as being "providing an error log analysis system that automatically generates diagnostic knowledge without the dependence of human experts such as field engineers."(col.2, line 25-28), and provides the solution to extract important parameters from case-based diagnostic system and finds applicable solution, however, does not provide any means to automatically generate corrective action execution as taught by Noble as stated above.

Therefore it would have been an obvious to one of an ordinary skill in art, having the teachings of Cuddihy and Noble in front of him at the time of invention was made, to include the teachings of Noble into the Cuddihy's case-based diagnostic system incorporating cases (higher level logs) such that the case-based diagnostic system automatically generates corrective action execution (triggering execution of commands) wherein the corrective routines (scripts- execution sets) are responsive or corrective processes initiated by the logs in response to the occurrence of a predefined event that be applied to multiple managed devices as prescribed by the system administrator (user).

This would have been obvious because Noble enhances the Cuddihy's system as it would not only automatically generate corrective action execution (triggering execution of commands) but also provides flexibility such as the corrective scripts (execution sets) can be run as remote shell execution on a management engine or management interface itself, one or more corresponding scripts which are responsive or corrective processes initiated by the alarm (status log) in response to the occurrence of a predefined event, and each event definition may be applied to multiple managed computers (devices), which boils down to providing a management system that is sufficiently flexible to handle the administration of a wide variety and a large number of managed computers (devices) in an efficient manner, essentially preventing the systems administrator from becoming overwhelmed with the increasing amount of management information and action required, and to automate these operations as much as possible, minimize the number of decisions and actions that are required by the system administrator, and a management system that is optionally self-configuring, as Noble puts it (col. 1, line 38-54).

Referring to claim 6,

Cuddihy teaches the network administration system of claim 1, (col. 3, line 53-59, "The sources include: field repair sites, central repair facility, quality control testing laboratories, etc. The plurality of error logs are then used as historical cases documenting software and hardware errors occurring at the different machines. Each of the error logs has a corresponding malfunction description (i.e. fault nz, yw, bd, etc.) associated with it."), wherein said means for receiving said status logs (col. 3, line 49-50, "The training unit 14 receives the plurality of error logs from various imaging devices located at different locations.") and generating higher level logs (col. 4, line 33-37, "After formatting, each of the plurality of error logs 30 are grouped in <u>the block finding and matching unit 32</u> into case sets of common symptoms or common corrective actions (i.e. faulty parts, boards, caches, etc.).") which satisfy predetermined rule sets (col. 4, line 33-50, "After formatting, each of the plurality of error logs 30 are grouped in the block finding and matching unit 32 into case sets of common symptoms or common corrective actions (i.e. faulty parts, boards, caches, etc.). FIG. 5A shows error logs 1-11 grouped into case sets, wherein error log cases 1-3 are grouped into case set I; error log cases 4-5 into case set II; error log cases 6-9 into case set III; and error log cases 10-11 into case set IV. A case is represented by one or more error logs and a fix associated with a single malfunction of a device. A case set consists of all historical cases having the same fix. After the error logs have been grouped into cases and case sets, the error logs in each case set are examined to identify common patterns of data. The common patterns are then labelled as blocks. A block consists of consecutive row(s) of data in a data file such as represented by FIGS. 4A-4B derived from a historical error log file that exists in at least one or more cases. Please note that error logs are status logs and "a case" is a "higher level log"); includes means for generating further higher level logs in response to receipt of at least one of: a) at least two different higher level logs; and b) at least one higher level log and at least one status log.(col. 7, line 60-64, "After the fault predicting unit 24 finds an applicable solution, a selector means 62 decides whether the new error log case 58 should be added to the historical cases (at least one higher level log and at least one status log) for use in diagnosing future malfunctions or discarded. In the present invention, the selector means adds cases to increase its accuracy in diagnosing future malfunctions.")

Referring to claim 7,

Cuddihy teaches method of claim 4, wherein providing rule sets includes providing rule sets that are satisfied (col. 3, line 49-50, "The training unit 14 receives the plurality of error logs from various imaging devices located at different locations.", col. 4, line 33-50, "After formatting, each of the plurality of error logs 30 are grouped in the block finding and matching unit 32 into case sets of common symptoms or common corrective actions (i.e. faulty parts, boards, caches, etc.). FIG. 5A shows error logs 1-11 grouped into case sets, wherein error log cases 1-3 are grouped into case set I; error log cases 4-5 into case set II; error log cases 6-9 into case set III; and error log cases 10-11 into case set IV. A case is represented by one or more error logs and a fix associated with a single malfunction of a device. A case set consists of all historical cases having the same fix. After the error logs have been grouped into cases and case sets, the error logs in each case set are examined to identify common patterns of data. The common patterns are then labelled as blocks. A block consists of consecutive row(s) of data in a data file such as represented by FIGS. 4A-4B derived from a historical error log file that exists in at least one or more cases. Please note that error logs are status logs and "a case" is a "higher level log"); by receipt of particular combinations of at least one of: a) at least two different higher level logs; and b) at least one higher level log and at least one status log (col. 7, line 60-64, "After the fault predicting unit 24 finds an applicable solution, a selector means 62 decides whether the new error log case 58 should be added to the historical cases (at least one higher level log and at least one status log) for use in diagnosing future malfunctions or discarded. In the present invention, the selector means adds cases to increase its accuracy in diagnosing future malfunctions.")

Conclusion

Examiner's note: Examiner has cited particular columns and line numbers in the references as applied to the claims above for the convenience of the applicant. Although the specified citations are representative of the teachings of the art and are applied to the specific limitations within the individual claim, other passages and figures may apply as well. It is respectfully requested from the applicant in preparing responses, to fully consider the references in entirety as potentially teaching all or part of the claimed invention, as well as the context of the passage as taught by the prior art or disclosed by the Examiner.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ashok B. Patel whose telephone number is (571) 272-3972. The examiner can normally be reached on 6:30 am-4:30 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Nathan A. Flynn can be reached on (571) 272-1915. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Ashok Patel

Ashok B. Patel
Examiner
Art Unit 2154

***